## 3.2 Instrukcja uruchomienia aplikacji przez programistę

Rozpoczęcie pracy nad kodem aplikacji wymaga zainstalowania na komputerze dewelopera środowiska programistycznego oraz serwera baz danych.

1. VisualStudio 2019

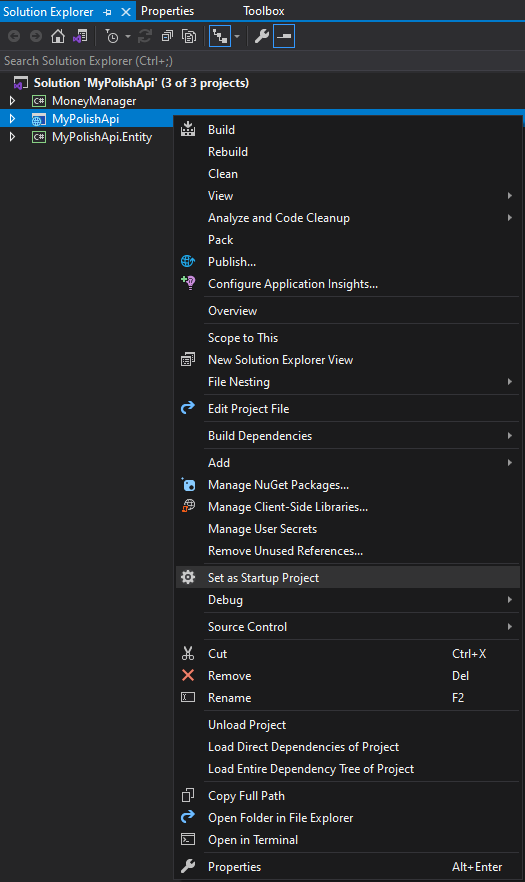
Minimalna wymagana wersja VS to wersja *Community*, która po rejestracji w witrynie Microsoft umożliwia tworzenie aplikacji w środowisku .NET bez konieczności wykupu licencji komercyjnej. Poprawne wczytanie solucji wymaga zainstalowania następujących komponentów:

1. opracowywanie zawartości dla platformy ASP.NET;
2. programowanie aplikacji klasycznych dla platformy .NET;
3. magazynowanie i przetwarzanie danych;
4. programowanie międzyplatformowe .NET.
5. SQL Server

W procesie projektowania MoneyManager wykorzystano serwer bazodanowy firmy Microsoft. Do poprawnego działania aplikacji wystarczy darmowa wersja Express. Aplikacja MoneyManager nie wymusza instalacji SQL Server i pozostawia kwestię wyboru serwera bazodanowego programiście. W przypadku używania innego narzędzia w trakcie konfiguracji aplikacji programista będzie musiał dostarczyć odpowiedni ConnectionString. Po spełnieniu wymagań wstępnych można przystąpić do konfiguracji solucji, co widoczne jest na *Rysunku 10*.

1. W VisualStudio otworzyć projekt MyPolishApi.sln
2. Ustawić projekt MyPolishApi jako startowy

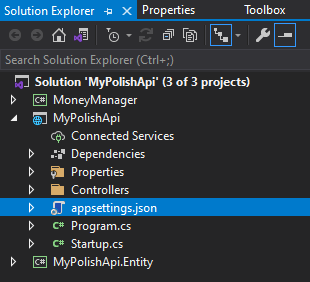
Rysunek 10. Ustawienie projektu MyPolishApi jako startowego.



Źródło: opracowanie własne.

1. Przejść do projektu MyPolishApi, a następnie otworzyć plik **appsettings.json**, którego lokalizację przedstawiono na *Rysunku 11.*

Rysunek 11. Lokalizacja pliku appsettings.json.



Źródło: opracowanie własne.

W pliku konfiguracyjnym należy ustawić konfigurację bazy danych poprzez dodanie sekcji ConnectionStrings o nazwie myLocalDb. Nazwa ta jest wykorzystywana w pliku Startup.cs i może być dowolnie zmieniana. Wymagane jest aby nazwy w obu plikach były takie same. Szczegóły sekcji ConnectionStrings przedstawiono na *Rysunku 12.*

Rysunek 12. Fragment kodu z sekcją ConnectionStrings.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Źródło: opracowanie własne.

Connection String powinien zawierać:

1. Server – nazwa komputera (lub adres IP) \ nazwa instancji (o ile podano w trakcie instalacji),
2. Database – nazwa bazy danych,
3. Trusted\_Connection=True – informuje o włączeniu Windows Authentication, szczegółowo przedstawiony na *Rysunku 13.*

Rysunek 13. Fragment kodu z Trusted\_Connection=True.

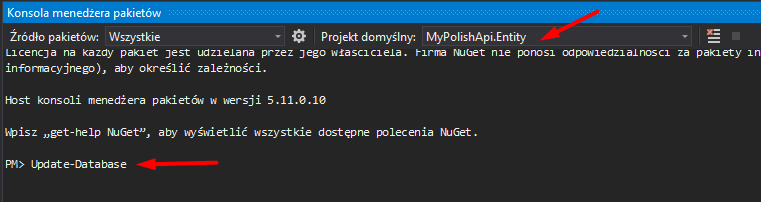
Obraz zawierający tekst

Opis wygenerowany automatycznie

Źródło: opracowanie własne.

1. Przejść do Konsoli Menadżera Pakietów w celu automatycznego stworzenia bazy danych. W menu **Widok** programu Visual Studio wybrać opcję **Inne Okna**, a następnie wybrać **Konsola Managera Pakietów**.
2. W konsoli wymienionej wyżej ustawić domyślny projekt na **MyPolishApi.Entity**, co zostało przedstawione na *Rysunku 14.* Ustawienie projektu startowego (jak opisano we wcześniejszych krokach), oraz domyślnego projektu w konsoli pakietów pozwoli Entity Framework Core na zlokalizowanie Connection Stringa, a następnie na połączenie się ze wskazaną w konfiguracji bazą danych. Po ustanowieniu połączenia Entity Framework Core wykona polecenia zawarte w migracji danych.

Rysunek 14. Konsola Managera Pakietów z MyPolishApi.Entity jako Projekt domyślny.



Źródło: opracowanie własne.

1. W ostatnim kroku wymaganym do uruchomienia aplikacji programista musi wskazać docelowy projekt, który będzie uruchomiony przez Visual Studio. W tym celu otwieramy właściwości solucji, których lokalizacja została zaznaczona na *Rysunku 15.*

Rysunek 15. Lokalizacja właściwości solucji.

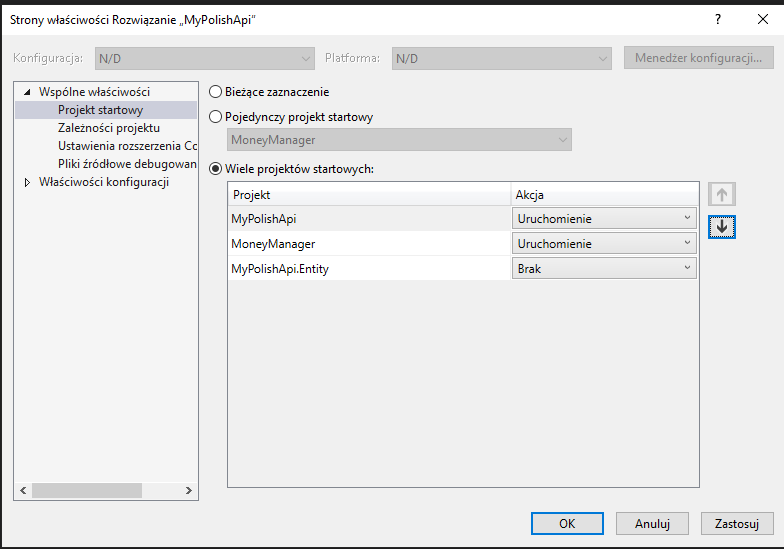
Obraz zawierający tekst

Opis wygenerowany automatycznie

Źródło: opracowanie własne.

Na koniec należy ustawić dwa projekty startowe zachowując przy tym odpowiednią kolejność jak pokazano na *Rysunku* *16*.

Rysunek 16. Kolejność projektów startowych.



Źródło: opracowanie własne.

Jak każda współczesna aplikacja, MoneyManager przechowuje konfigurację użytkownika w pliku user.config w katalogu AppData w lokalnym katalogu użytkownika przedstawionym na *Rysunku 17.*

Rysunek 17. Lokalizacja pliku user.config.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Źródło: opracowanie własne.

W Konfiguracji pokazanej na *Rysunku 18*, zapisywane są ustawienia użytkownika takie jak: budżet oraz nazwy kategorii przypisywane transakcjom.

Rysunek 18. Konfiguracja

Obraz zawierający tekst

Opis wygenerowany automatycznie

Źródło: opracowanie własne